



# Noise-Adaptive Shape Reconstruction from Raw Point Sets

Simon Giraudot, David Cohen-Steiner, Pierre Alliez

## ► To cite this version:

Simon Giraudot, David Cohen-Steiner, Pierre Alliez. Noise-Adaptive Shape Reconstruction from Raw Point Sets. Computer Graphics Forum, 2013, 32 (5), pp.229-238. 10.1111/cgf.12189 . hal-00844472

**HAL Id: hal-00844472**

**<https://inria.hal.science/hal-00844472>**

Submitted on 15 Jul 2013

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Noise-Adaptive Shape Reconstruction from Raw Point Sets

Simon Giraudot

David Cohen-Steiner

Pierre Alliez

Inria Sophia Antipolis — Méditerranée

---

## Abstract

*We propose a noise-adaptive shape reconstruction method specialized to smooth, closed shapes. Our algorithm takes as input a defect-laden point set with variable noise and outliers, and comprises three main steps. First, we compute a novel noise-adaptive distance function to the inferred shape, which relies on the assumption that the inferred shape is a smooth submanifold of known dimension. Second, we estimate the sign and confidence of the function at a set of seed points, through minimizing a quadratic energy expressed on the edges of a uniform random graph. Third, we compute a signed implicit function through a random walker approach with soft constraints chosen as the most confident seed points computed in previous step.*

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Boundary representations.

---

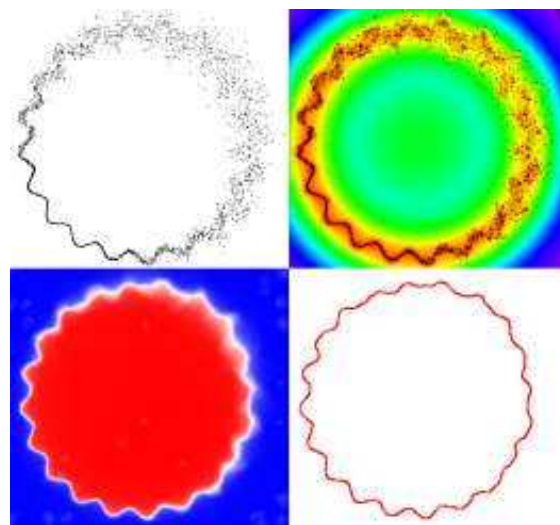
## 1. Introduction

The increasing variety of sensors for acquiring point sets corresponds to a range of defects inherent to each sensor and associated acquisition process. The point sets may differ in terms of sampling (density, anisotropy, missing data), noise and outliers. In addition, the level of noise may vary within the same point set, depending on the type of noise (uncertainty of sensor device, registration), acquisition condition and light-material interaction. One example is the Kinect sensor, where noise and depth are correlated [NIL12]. Our quest for robustness includes the ability to deal with variable noise. This motivates a reconstruction method which automatically trades smoothness on noisy areas, for faithfulness to input point set on noise-free areas.

### 1.1. Related Work

This paper tackles the shape reconstruction problem from raw point sets, with focus on shapes that are both smooth and closed, and on robustness to variable noise, outliers and missing data. We restrict our review of previous work to reconstruction approaches that share at least one aspect of our focus.

**Noise.** Robustness to noise is commonly handled through smoothing, integral computations, variational formulations



**Figure 1:** Algorithm overview. Top: input point set with variable noise, and noise-adaptive distance function to the inferred shape. Bottom: sign guess and confidence, and reconstructed shape.

or scale-space approaches. The popular Poisson surface reconstruction approach involves smoothing and a variational

(AUTHORS COPY)

Computer Graphics Forum © 2013 The Eurographics Association and Blackwell Publishing Ltd. Published by Blackwell Publishing, 9600 Garsington Road, Oxford OX4 2DQ, UK and 350 Main Street, Malden, MA 02148, USA.

formulation through the Laplace operator [KBH06]. Amenta and Kil define a point set surface where robustness to noise is achieved through Gaussian weights [AK04]. Digne use a scale space to render the interpolant reconstruction problem better posed [Dig10]. Dey and Goswami rely on the property of Delaunay triangulations of noisy point sets [DG06].

**Outliers.** Robustness to outliers has also been investigated through outlier removal [Sot06], data clustering [Son10], robust norms such as the  $l_1$ -sparse norm [ASGC010], spectral methods [KSO04], or robust distances [CCSM11].

**Missing data.** For the reconstruction of closed surfaces, robustness to missing data is commonly tackled through implicit formulations. The latter often include a smoothness assumption in order to make the hole-filling problem better posed. One step further is to estimate the shape and structure of missing data. Shalom et al. fit visibility cones in order to estimate the outside space visible from the acquisition device [SSZCO10]. Tagliasacchi et al. devised a volume-aware surface evolution approach which trades visibility for surface and volume smoothing [TOZ\*11]. Berger and Silva introduced the notion of medial kernels in order to recover the structure of missing data [BS12].

**Variable noise.** The noise-adaptive facet of the shape reconstruction problem has received less attention. Wang et al. proposed an automatic bandwidth selection method for moving least squares (MLS) surfaces [WSS09]. Unnikrishnan et al. proposed a scale selection method that relates the local geometry of a shape to a statistical estimator such as principal component analysis [ULVH10]. The main rationale behind their adaptive scale parameter is to minimize the estimation error. Mellado et al. define a so-called adaptive bandwidth through scale space analysis [MBG\*12], the rationale being to progressively lower the scale as long as the regularity of the local geometric variation stays below a user-specified threshold.

## 1.2. Contributions and Overview

Our first contribution is a novel noise-adaptive robust distance function which relies on the only assumption that the inferred shape is a smooth submanifold of known dimension. Our second contribution is a graph-based approach to guess the sign of this function, with resilience to missing data.

Our algorithm takes as input a raw point set (without requiring normals) sampling the boundary of the inferred solid object. The algorithm comprises three main steps (Figure 1):

1. **Distance:** We compute a noise-adaptive unsigned distance function to the inferred shape and represent it on a non-uniform triangulation obtained through Delaunay refinement (§3.1).
2. **Sign guess:** We estimate the sign of the signed distance function to the inferred shape, at the vertices of a regular grid. The sign estimates – and associated confidences

– are obtained through constructing a uniform random graph formed by edges connecting pairs of grid vertices, and minimizing a quadratic energy related to the sign difference estimate between the edge vertices (§3.2).

3. **Reconstruction:** We compute the final signed distance function on the non-uniform triangulation through a random walker approach. The latter minimizes a Dirichlet energy locally weighted by the scale of the noise-adaptive distance function, and softly constrained by the most confident sign guesses computed in previous step (§3.3).

Before detailing the main steps of the algorithm we provide next some details on the noise-adaptive distance function.

## 2. Background

Chazal et al. [CCSM11] introduced the notion of robust distance function from a query point  $x$  to a probability distribution  $\mu$  in  $\mathbb{R}^n$ . Denoting by  $r_{\mu,m}(x)$  the minimal radius  $r$  such that the ball centered at  $x$  with radius  $r$  encloses a mass of at least  $m$ , the robust distance function is defined by:

$$d_{\mu,m}^2 : \mathbb{R}^n \rightarrow \mathbb{R}, x \mapsto \frac{1}{m} \int_{B(x,r_{\mu,m}(x))} \|x-y\|^2 d\mu(y).$$

The user-defined parameter  $0 < m \leq 1$ , when taking the input point set as a discrete distribution of  $n$  point masses, specifies the number  $K = mn$  of nearest neighbors involved in the above formula. It was shown that the sublevel sets of this distance provide a topologically (more precisely, homotopically) accurate approximation of the surface to be reconstructed under suitable sampling conditions that allow for noise and outliers. This inference result only depends on two properties of the robust distance, namely its robustness in the Wasserstein distance, and the 1-semiconcavity of its square.

From a practical point of view, these distance functions have proved relevant for noise- and outlier-robust surface reconstruction [MDGD\*10] but require a trial-error process to select the scale parameter  $K$  that provides a means to trade robustness for accuracy. Furthermore, a satisfactory global choice for  $K$  can not be found with variable noise. We propose next a noise-adaptive variant.

Our approach for defining adaptive distances is based on the sole assumption that the inferred shape is a smooth submanifold of known dimension  $k$  (the practically interesting cases being  $k = 1$  or  $k = 2$ ). Consider an input measure  $\mu$  and fix a constant parameter  $\alpha > 0$ . We define the adaptive distance function as follows:

$$\delta_{\mu,\alpha} = \inf_{m>0} \frac{d_{\mu,m}}{m^\alpha}.$$

To see how this function behaves, consider an ideal case with a uniform continuous measure  $\mu$  on a  $k$ -subspace in  $d$ -dimensional space. At distance  $h$  from the subspace, an easy

calculation shows that:

$$d_{\mu,m}^2(h) = c_1 m^{2/k} + h^2,$$

where  $c_1$  depends on  $k$  and the density of  $\mu$ . Hence for  $\alpha < 1/k$ ,  $d_{\mu,m}(h)/m^\alpha$  is unimodal as a function of  $m$ , and reaches its minimum at a value  $m^*$  given by:

$$m^*(h) = c_2 h^k$$

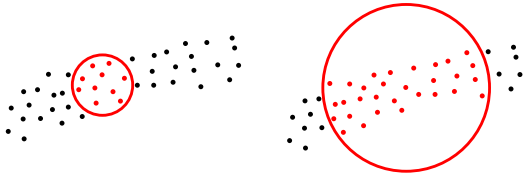
We thus obtain:

$$\delta_{\mu,\alpha}(h) = c_3 h^{1/k-\alpha}.$$

We see a first difference in the behavior of the adaptive distance and the robust distance. While the robust distance is a smooth quadratic function that does not reach zero on the data, the modified distance function does vanish on the data, and grows rapidly (with a vertical tangent) as we move away from it. In particular, the level sets of the adaptive distance accumulate tightly around the data, giving a more precise localization of the underlying manifold.

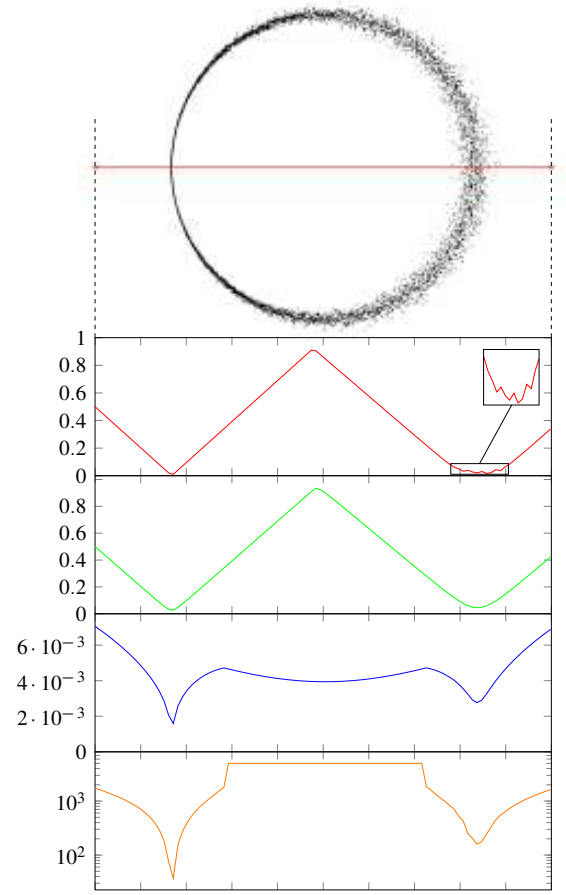
Consider now the case of an ambient noise in  $d$ -space. Then  $d_{\mu,m}^2 = c_1 m^{2/d}$ . So for  $\alpha > 1/d$ , we get that  $d_{\mu,m}/m^\alpha$  is decreasing with  $m$ , implying  $m^* = 1$ , and  $\delta_{\mu,\alpha}^2$  is, in the discrete case, the average squared distance to all data points.

As shown in Figure 2, the case of a noisy  $k$ -submanifold can be seen as a mixture of the two previous cases. Consider a point  $x$  lying on the unknown submanifold. At a scale smaller than the noise level, the data looks like ambient noise, meaning that for small values of  $m$ ,  $d_{\mu,m}(x)/m^\alpha$  is decreasing, provided that  $\alpha > 1/d$ . As soon as the scale gets larger than the noise level, the data starts looking like a  $k$ -submanifold, hence  $d_{\mu,m}(x)/m^\alpha$  starts increasing, assuming  $\alpha < 1/k$ . Hence, under these conditions, the minimum is reached for a value  $m^*$  that adjusts to the local noise level. As a consequence, the adaptive distance function gives an accurate representation of the data where the sampling quality is good, while smoothing the data in areas where geometric details are lost due to poor sampling quality. We note that these properties depend on the fact that  $1/d < \alpha < 1/k$ . In all experiments we chose  $\alpha = 3/4$  for curves in 2D and  $\alpha = 5/12$  for surfaces in 3D.



**Figure 2: Apparent dimension.** Left:  $K = 10$ , the apparent dimension is 2 (a surface). Right:  $K = 30$ , the apparent dimension is 1 (a curve).

From a more theoretical perspective, we remark that the adaptive distance function may be modified so as to satisfy the Wasserstein robustness and semiconcavity proper-

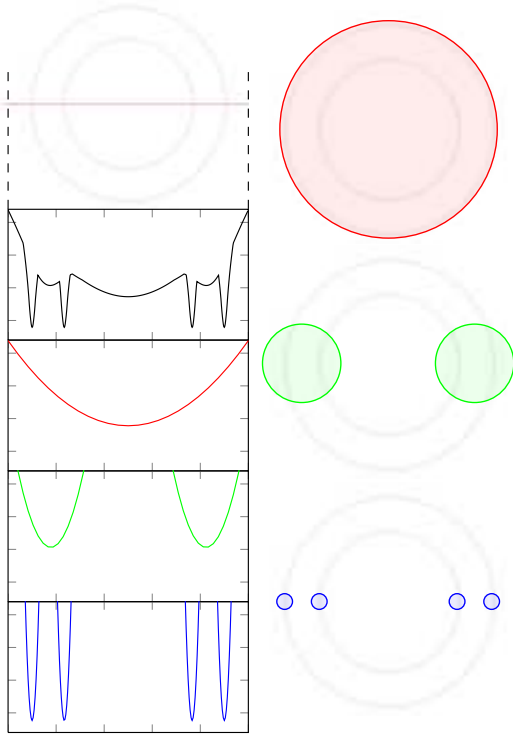


**Figure 3: Distance functions.** Top: input point set and segment selected to depict function values. **Red curve:** robust function  $d_\mu$  with  $K = 6$ : small details are captured in noise-free area, but the function is noisy on noisy area. **Green curve:** robust function  $d_\mu$  with  $K = 70$ : noisy areas are captured, but noise-free areas are over-smoothed and the function first minimum has shifted to the right. **Blue curve:** adaptive function  $\delta_\mu$ : all features are captured. **Orange curve:** selected value for  $K$ : notice the high dynamic of the function (log vertical scale). The flat maximum appears when the total number of points is reached.

ties that are needed for correct topological inference. Define the modified function using the same principle but limiting the infimum over values of  $m$  that exceed a threshold  $m_0$ . For each value of  $m$ , the robust distance  $d_{\mu,m}$  is  $1/\sqrt{m}$ -robust, which means that two measures  $\mu$  and  $\mu'$  that are  $\epsilon$  away in the Wasserstein 2-distance will have robust distances  $d_{\mu,m}$  and  $d_{\mu',m}$  at most  $\epsilon/\sqrt{m}$  away in the sup norm. Its square is also 1-semiconcave. So all functions in the infimum are  $m_0^{-\alpha-1/2}$ -robust and with  $m_0^{-2\alpha}$ -semiconcave squares. Since these properties are preserved under taking infimum, so does the modified distance. In practice we actually im-

pose such a lower bound on  $m$  by taking at least 6 nearest neighbors.

As discussed above, the scale selected by the infimum is roughly speaking the smallest at which the data looks at most  $1/\alpha$ -dimensional. One difficulty is that for certain shapes several scales may coexist, with the risk of blurring the geometric information. In Figure 4 for example, we see that the adaptive distance remains relatively low between the two circles because these look like one from a distance. One way of resolving this issue is to impose an upper bound on  $m$  in the infimum. Indeed, the scale at which the two circles merge corresponds to a high value for  $m$ , hence by setting an appropriate upper bound we would observe a sharp increase of the adaptive distance, similar to the last curve in the figure. Another option is to focus on the smallest scale found, by detecting the sharpest peaks in the graph of the distance. We will use both ideas in the sequel.

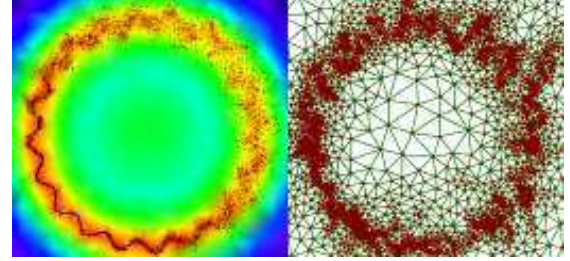


**Figure 4:** Adaptive function  $\delta_U$  on a line segment and its scale decomposition. **Top:** input point set and line segment chosen to depict the function values. **Black curve:**  $\delta_U$ : all features are captured. The curve may be seen as an infimum of the following curves (corresponding scales shown right). **Red curve:** large scale ( $K$  equating the total number of input points). Far from the points, the inferred shape is seen as a point object at the center of mass of the points. **Green curve:** intermediate scale. Between the 2 circles of the ring, the shape is seen as a single noisy circular shape. **Blue curve:** smallest scale. The shape is seen as the noise-free ring sought after.

### 3. Algorithm

#### 3.1. Distance

We approximate the noise-adaptive distance function through piecewise linear interpolation on an isotropic triangulation. For improved scalability we require the triangulation to be adaptive, with high density of vertices where the function gradient varies rapidly. The triangulation is obtained through Delaunay refinement of a coarse triangulation initialized with the loose convex hull of the input points. The triangulation is greedily refined by inserting Steiner vertices into the triangulation [RY07] until all cells are considered good for two criteria: shape and function interpolation error (Figure 5). The shape criterion relates to the radius-edge ratio (circumradius to shortest edge length). The interpolation error criterion relates to a user-specified maximum error  $e_{max}$  between the distance function  $\delta_\mu$  and its linear interpolation onto the cell. To make the error evaluation computationally tractable we probe each cell with a finite number of probes, bounded by a user-specified maximum  $n_{max}$ .



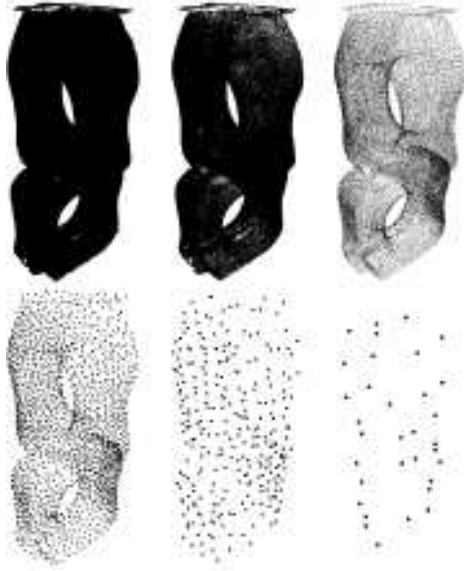
**Figure 5:** Delaunay refinement. Left: input point set and distance function. Right: triangulation computed with  $n_{max} = 15$  and  $e_{max} = 0.001$ .

A scalability issue comes from the fact that  $\delta_\mu$  is defined as a minimum search over all possible  $K$  values of a  $K$  nearest neighbor search. The latter involves a  $k$ -d tree structure. For a single query search in a point set of size  $N$ , the time needed is proportional to  $O(K \log(N))$ . For all search queries with  $K$  between 1 and  $N$  the complexity is  $O(N \log(N))$ . To alleviate this issue we construct several  $k$ -d trees through hierarchical clustering of the input point set:

1. In addition to the original point set, we store clustered versions of it obtained through hierarchical clustering [PGK02], parameterized to be uniform, i.e., with no constraints on the local variation (Figure 6). Each of these point sets gets the index  $i$ , with index 0 for the input point set. Denote by  $s$  the size of each cluster (set by default to 10 in all shown experiments), the size of each point set of index  $i$  is  $\frac{N}{s^i}$ .
2. We construct a  $k$ -d tree for each of these point sets, where the first  $k$ -d tree with index 0 contains the input point set, and all subsequent trees contain cluster points that are centroids obtained through hierarchical clustering.
3. To evaluate the robust distance function  $\delta_\mu$  for each query

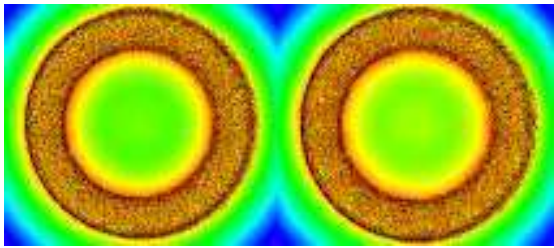


point with parameter  $K$ , we use in sequence these  $k$ -d trees with increasing  $K$  until reaching the desired  $K$ . More specifically, we approximate the distance  $\delta_\mu$  to the  $K$  nearest neighbors by increasing  $K$  and summing up squared distances to cluster points with appropriate weights  $s^i$ . When  $K$  gets larger than  $s^i$ , we query the next  $k$ -d tree with index  $i + 1$ . The complexity for a single query drops to  $O(\log(N)^2)$ .



**Figure 6: Multiscale  $k$ -d tree.** The input point set (1M points) is hierarchically clustered with clusters of size 10.

The main rationale behind this approximation is that the function  $\delta_\mu$  computed through minimum search over  $K$ , is more and more regular with increasing  $K$ . We thus need an accurate computation for small  $K$  through the  $k$ -d tree of the input point set, and can rely on coarser approximations for larger  $K$ . This approach experimentally provides satisfactory approximations of  $\delta_\mu$  with a substantial gain in complexity (Figure 7).



**Figure 7: Approximating  $\delta_\mu$ .** The 3D point set (50K points) is sampling a torus with variable noise. The functions are depicted on a planar slice. Left: exact computation (3 minutes). Right: approximation with a multi-scale  $k$ -d tree (1 second).

### 3.2. Sign Guess

The final output of our algorithm is a signed implicit function whose zero isolevel defines the reconstructed shape. Signing the noise-adaptive function  $\delta_\mu$  as performed by Mullen et al. [MDGD\*10] is not appropriate for two main reasons: i) values of  $\delta_\mu$  depend on the local amount of noise and point density, hence selecting a global value for the  $\epsilon$ -band is impossible, ii) isolevels of  $\delta_\mu$  on areas with missing data are in general not aligned with a smooth, plausible way to fill the holes. Instead, we first aim at guessing the sign (inside or outside the inferred object) and confidence in the sign, in order to determine reliable seed points that are used as constraints in the subsequent step.

We propose a novel approach based on a random graph  $G$  constructed as follows:

- Nodes are placed at the vertices of a regular grid that covers the domain  $\Omega$  of the input point set, defined as its loose bounding box.
- Edges are generated by randomly picking connecting pairs of nodes. Each edge is assigned the sign attribute  $-1$  if its two end nodes are estimated to have the same sign,  $+1$  otherwise.

On this graph we define and minimize the following energy:

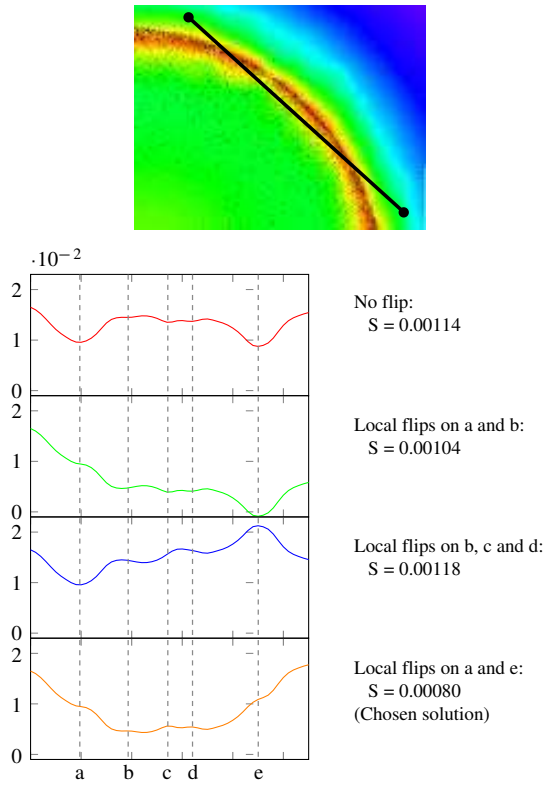
$$E_G(f) = \sum_{(i,j) \in G} (f_i + \epsilon_{i,j} f_j)^2,$$

where  $(i, j)$  denotes an edge between two nodes  $i$  and  $j$ ,  $f_i$  denotes the sign-guess function at node  $i$ , and  $\epsilon_{i,j}$  denotes the edge sign attribute. Notice that this approach may be seen as a generalization of previous work based on ray shooting [MDGD\*10] as a ray amounts to pick an edge of the graph connected to the boundary of the domain. Our approach randomly selects edges without any length condition within the domain, the number of edges selected being our means to trade robustness for computational time.

Determining the attribute of an edge requires estimating the number of crossings between the edge and the inferred shape. The unsigned function  $\delta_\mu$  locally drops near the inferred shape, but a simple thresholding is not satisfactory as the function value depends on the local noise level and point density. We rely instead on a smoothness assumption: among all possible ways to sign an unsigned distance function, we select the one that leads to the smoothest signed function. However, the function  $\delta_\mu$  does not reach 0 on the inferred shape, and the number of ways to sign it is infinite as it is a continuous function. For these reasons we consider candidate sign flips only at the local minima of  $\delta_\mu$ , and replace the notion of sign flips by *local flips* that mirror the function with respect to a horizontal line placed at these local minima. The local sign guess problem for an edge thus becomes a combinatorial problem of complexity  $2^N$ , where  $N$  denotes the number of local minima on the edge. We select the combination of local flips that yields the smoothest signed func-

tion (Figure 8). The attribute  $\epsilon_{i,j}$  is set to  $-1$  if the number of local flips performed is even, and to  $1$  otherwise.

To measure smoothness of each signed function during combinatorial search we use the squared norm of its second derivative. The latter is measured and summed up at multiple scales, and computed through finite differences after uniform discretization along the edge. To accelerate computations we i) evaluate the piecewise linear interpolation of the unsigned function on the Delaunay triangulation constructed in first step, ii) perform before minimum detection a mollification of the function along the edge through Gaussian filtering to avoid the spurious local minima that arise on noisy areas, and iii) ignore the local minima that occur at areas where the optimal scale  $K$  value is very high (500 in all examples shown). Constructing a graph of 300K nodes and 20M edges takes less than 3 minutes.

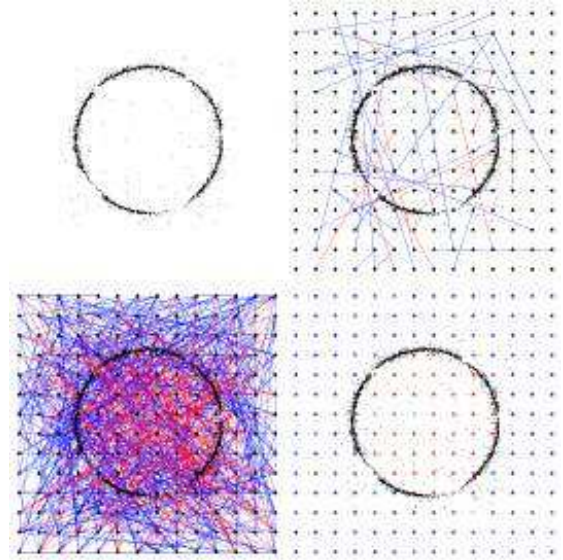


**Figure 8: Signing through local flips at local minima.** Top: 5 minima found on the edge. Top curve (red): no flip and corresponding smoothness value ( $S$ ). Bottom curve (orange): the chosen combination of local flips corresponds to the smoothest curve.

Each edge is seen as a signing hypothesis, more or less local depending on the edge length. We then build a global consensus from these hypotheses by minimizing  $E_G$ . To avoid the trivial solution where  $f$  is the zero function we

need to constrain the signed function. The most natural constraint would be to impose that the solution has unit  $L_2$  norm, but this leads to an eigenvalue problem with limited scalability. Another option is to constraint domain boundary node values to 1, but this cannot handle situations where the object is partially scanned, since part of the bounding box boundary is then inside the object. We found that constraining the solution to have average value equal to 1 achieved excellent results in all situations, while requiring only a linear solve.

After solving we determine for each node  $i$  its confidence  $c$  in the sign as the ratio of edges  $e_{i,j}$  adjacent to  $i$  whose coefficient  $\epsilon_{i,j}$  is in agreement with the signs of  $f_i$  and  $f_j$ . All nodes with confidence higher than a defined threshold ( $c > c_{min}$  with  $c_{min} = 0.75$ ) are referred to as *confident nodes* and used as constraints in the next step. As linear solver we use a conjugate gradient algorithm applied to a sparse matrix, using the Eigen library [GJ\*10]. Solving for a graph with 300K nodes and 20M edges takes 10 seconds. The random edge selections and attribute computations are parallelized. Figure 9 illustrates the random graph in 2D.



**Figure 9: Random graph.** Top: input point set and edges of the graph (only 1% of edges are shown for clarity, with blue for similar signs and red edges for different signs). Bottom: 20% of the graph edges shown, and signed function at graph nodes after linear solve (red for inside, blue for outside).

### 3.3. Reconstruction

Given the set of confident nodes computed in previous step, we now wish to compute a signed implicit function  $g$  defined on the adaptive triangulation  $T$ , whose 0 isovalue defines the reconstructed shape.

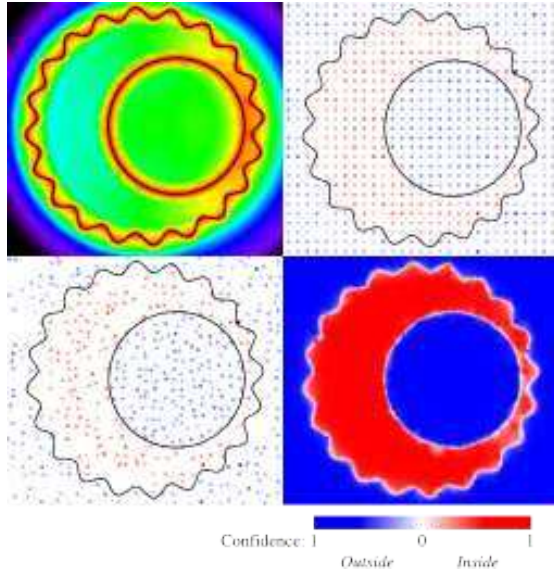
Our method is inspired by the random walker approach

used for image segmentation [Gra06]: we search for a function  $g$  that minimizes the following weighted Dirichlet energy:

$$E_{g,T} = \int_{\Omega} w(x) |\nabla g(x)|^2 dx,$$

where  $w(x)$  is chosen as follows:  $w(x) = \delta_{\mu}(x)$  if  $K(x)$  is lower than a large value ( $K_{max} = 500$  in all experiments) and  $w(x)$  is set to an arbitrarily large value else. Solving for this energy without any constraint would produce the trivial function  $g = 0$ . We thus use as soft constraints the confident nodes assigned to the closest vertices of  $T$ . The rationale is to compute an approximate indicator function of the inferred shape where the function varies abruptly at the inferred shape.

Solving for this energy in the space of piecewise-linear functions defined on  $T$  boils down to solving the linear system  $(L + \alpha C)X = \alpha B$ , where  $L$  is a weighted Laplacian matrix (size  $V \times V$ , where  $V$  is the number of vertices of  $T$ );  $\alpha$  is a user-specified coefficient used to weight the influence of constraints ( $10^{-3} \times \text{trace}(L)/V$  in experiments),  $C$  is a diagonal matrix with  $c_{i,i} = 1$  if vertex  $i$  is constrained, 0 otherwise;  $X$  is the solution vector solved for;  $B$  is the right hand side vector where  $b_i = 0$  if vertex  $i$  is unconstrained, and  $b_i = -1$  or  $b_i = 1$  depending on the sign constraint otherwise (Figure 10).



**Figure 10: Signed implicit function.** Top: unsigned function and sign guess on nodes of uniform graph. Bottom: confident nodes used as soft constraints and signed implicit function after linear solve.

#### 4. Experiments

In 2D, the final reconstructed curves are obtained through marching triangles, which contours the 0-level of the signed

implicit function represented on the 2D adaptive triangulation. In 3D, the final reconstructed surfaces are obtained through meshing the 0-level of the signed implicit function represented on the 3D adaptive triangulation. Meshing is achieved through Delaunay refinement [RY07] instead of a marching-tetrahedra approach which generates overly complex meshes.

As a sanity check we first apply our algorithm on a low-noise point set generated from photo sensors (courtesy EPFL Computer Graphics and Geometry Laboratory [112]). Figure 11 depicts a point set (little noise and few holes due to missing data), our reconstruction and a comparison with the popular Poisson surface reconstruction method [KBH06] using the normal vectors provided with the original point set. Our experiments show that our approach produces comparable results, with different behavior for hole filling (an ill-posed problem).



**Figure 11: Low noise.** From left to right: raw point set; point set & reconstruction; reconstruction only; Poisson reconstruction.

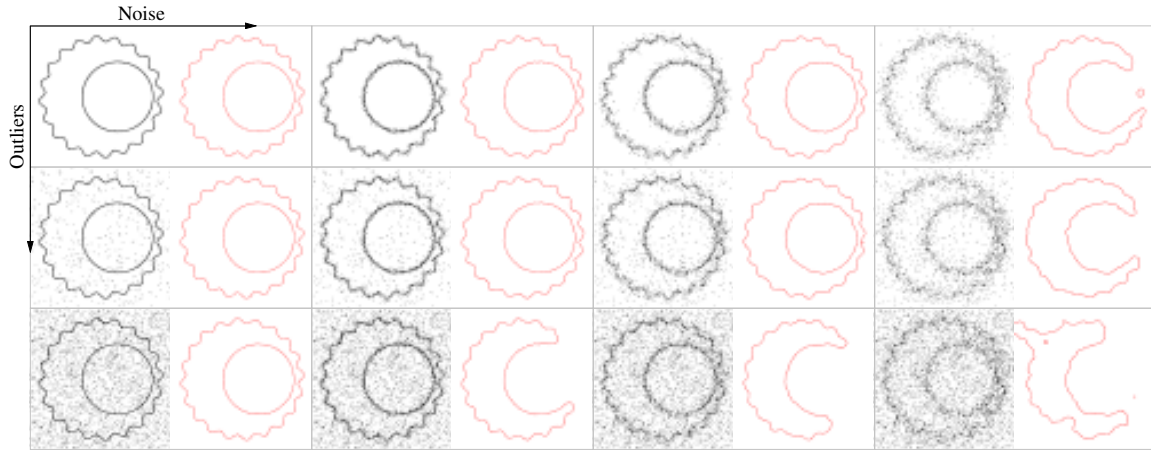
The cases where our algorithm starts making an added value are the ones either with noise and outliers, or when the estimation of oriented normals is impossible. Figure 12 illustrates the stability of our approach (until failure) against an increasing amount of noise and outliers.

The cases where our algorithm really makes an added value are the ones with variable noise. We illustrate resilience to gradually variable noise in Figure 1, where the noise is added with a linear increase. Figure 13 illustrates the fact that noise-free areas are over-smoothed when using a non-adaptive function with a constant scale parameter  $K$ , set to a sufficiently large value to get robustness on noisy area.

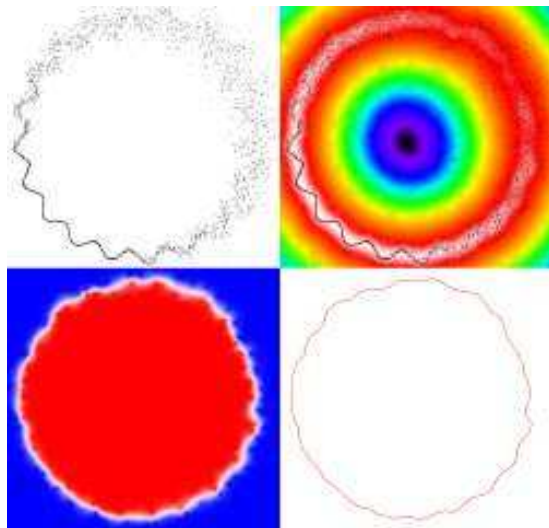
Resilience to gradually variable noise is illustrated in 3D by Figure 14: the reconstructed surface smoothly approximates the inferred shape on noisy area while providing high accuracy on noise-free area. Another strength of our approach is the absence of shrinkage in the case of noisy areas with non-zero curvature (Figure 15).

We illustrate in Figures 16 and 17 the surfaces recon-





**Figure 12: Noise and outlier robustness.** The noise increases from left to right. The outliers increase from top to bottom, ranging from outlier-free to 60% through 20%. Input point set in black, reconstructed curve in red.

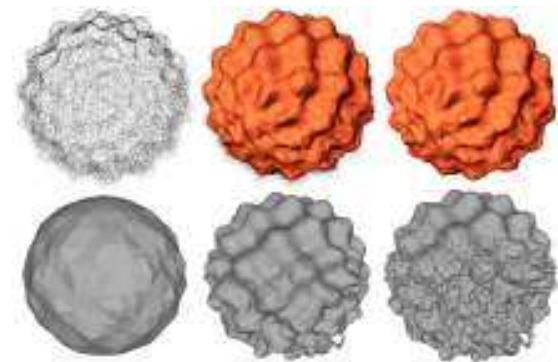


**Figure 13: Reconstruction with a constant scale ( $K = 80$ ).** Top: input point set with variable noise, and unsigned distance function to the inferred shape. Bottom: sign guess and confidence, and reconstructed shape, over-smoothed.

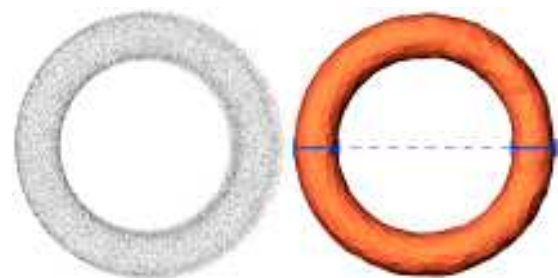
structed from point sets with 2 distinct levels on noise. Such noise appears in application scenarios where various devices or acquisition conditions have been used for each object or part of the object.

Figure 18 pushes our algorithm to its limit with a point set obtained through dense photogrammetry: with variable noise, outliers as well as structured outliers.

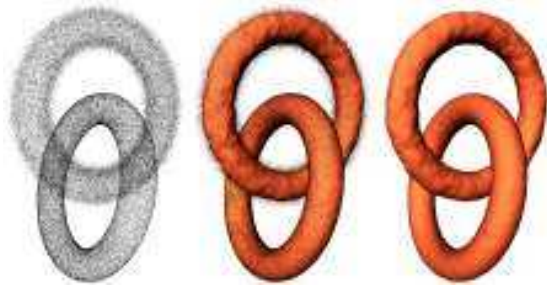
As our theory is based on a dimension assumption, we also experiment with cases close to failure where the noise level is so high that the dimension of the underlying shape is ambiguous (Figure 19). Notice how the geometry is altered while the topology is still captured.



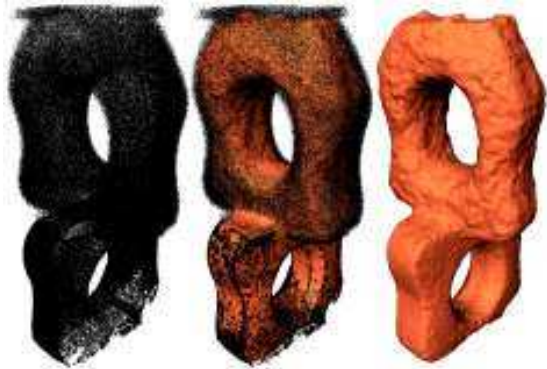
**Figure 14: Gradually variable noise (generated).** Top: raw point set, where noise increases linearly from top to bottom; point set & our reconstruction; our reconstruction only. Bottom: Poisson reconstruction with a constant octree depth of 4, 6 and 8.



**Figure 15: Variable noise (generated).** Left: top view of input point. Right: top view of reconstructed surface. Notice that the reconstructed surface exhibits no shrinkage.



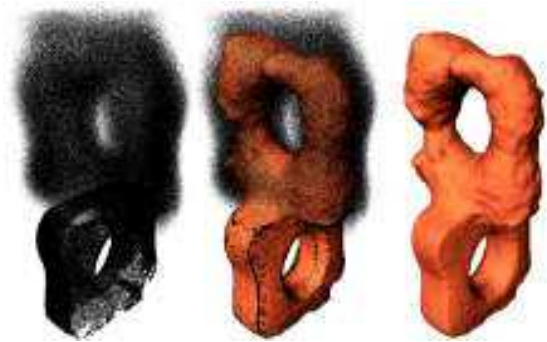
**Figure 16: Two levels of noise (generated).** Left: raw point set containing a noise-free and a noisy torus. Middle: point set & reconstruction. Right: reconstruction only.



**Figure 17: Two levels of noise.** Left: raw point set with additional noise on the top half part. Middle: point set & reconstruction. Right: reconstruction only.



**Figure 18: Noise and structured outliers.** Left: raw point set. Middle: closeup on point set. Right: closeup on point set & reconstruction.



**Figure 19: Noise almost beyond dimension assumption.** Left: raw point set with high noise on the top half part. Middle: point set & reconstruction. Right: reconstruction only.

Figure 20 illustrates a failure case where the noise truly exceeds the dimension assumption.



**Figure 20: Noise beyond dimension assumption (generated).** Left: raw point set. Middle: point set & reconstruction. Right: reconstruction only.

Finally, our framework is resilient to noise and outliers, but not to widely variable sampling density as low density areas are considered outliers (Figure 21).



**Figure 21: Variable density (generated).** Left: raw point set. Middle: point set & reconstruction. Right: reconstruction only. Our method fails in capturing the correct dimension in the low density area.

Table 22 provides timings and memory consumptions.

#### 4.1. Limitations

Using a uniform grid for the nodes of the graph requires a large number of nodes to capture the correct topology of shapes with small feature size due to small separation or thickness. This obviously leads to scalability issues. We experimented with a non-uniform graph where the nodes are reusing the vertices of the adaptive triangulation used in step 1, and where appropriate weights per edge are devised to compensate for the non-uniformity. None of these experiments led to satisfactory results.

For computing the sign attribute of an edge of the graph, we use an exhaustive combinatorial search in the number of (retained) local minima. In our experiments the number of retained minima is on average below 6, but for complex shapes with many sheets this can also lead to scalability issues.

Finally, we are using a two-step approach for guessing the sign, then solving for the signed implicit function. Our approach is scalable as involves only linear solves, but it would be more consistent to do everything in one step without hampering the scalability.

Point set	Size	T <sub>distance</sub>	T <sub>sign_guess</sub>	T <sub>reconstruction</sub>	T <sub>total</sub>	Memory	Specific parameters
Figure 11	857.726	207	32	12	242	700 MB	
Figure 14	50.000	52	25	4	82	300 MB	
Figure 16	50.000	91	33	6	130	400 MB	$K_{max} = 1000$
Figure 18	419.488	181	155	6	342	1.6 GB	$e_{max} = 0.003$ , 500K nodes, 15M edges, $l = 0.60$

**Figure 22: Timings.** Timings in seconds. Default parameters:  $n_{max} = 10$ ,  $e_{max} = 0.004$ , 50K nodes, 1.5M edges,  $c_{min} = 0.75$ ,  $K_{max} = 500$ .

## 5. Conclusion

We have presented a shape reconstruction method specialized to smooth, closed shapes. Our main contribution is a novel robust unsigned distance function which provides resilience to variable noise through automatic local scale selection. The main rationale behind this function is to assume that the inferred shape is a smooth submanifold of known dimension. We then leverage this robust function into a new shape reconstruction algorithm that computes a signed implicit function, with resilience to variable noise, outliers and missing data. This algorithm only involves solving two linear systems on sparse matrices.

We also observed that the robust unsigned function provides a means to reveal the different scales of the shape. This motivates future work on multi-scale or hierarchical shape reconstruction, as well as automatic scale selection for geometry processing.

## Acknowledgements

This work was funded by the European Research Council (ERC Starting Grant “IRON: Robust Geometry Processing”, Grant agreement 257474). We wish to thank David Bommes for advice on linear and non-linear solvers.

## References

- [112] EPFL Computer Graphics and Geometry Laboratory, EPFL statue model repository. <http://lgg.epfl.ch/statues.php?p=dataset>, 2012.
- [AK04] AMENTA N., KIL Y.: The domain of a point set surface. In *Proceedings of EUROGRAPHICS conference on Point-Based Graphics* (2004), pp. 139–147.
- [ASGCO10] AVRON H., SHARF A., GREIF C., COHEN-OR D.: L1-sparse reconstruction of sharp point set surfaces. *ACM Transactions on Graphics* 29 (November 2010), 135:1–135:12.
- [BS12] BERGER M., SILVA C.: Medial kernels. *Computer Graphics Forum* 31 (2012), 795–804.
- [CCSM11] CHAZAL F., COHEN-STEINER D., MÉRIGOT Q.: Geometric inference for probability measures. *Foundations of Computational Mathematics* 11, 6 (2011), 733–751.
- [DG06] DEY T. K., GOSWAMI S.: Provable surface reconstruction from noisy samples. *Comput. Geom. Theory Appl.* 35, 1 (Aug. 2006), 124–141.
- [Dig10] DIGNE J.: *Inverse Geometry: From the raw point cloud to the 3D Surface (Theory and Algorithms)*. PhD thesis, Ecole normale supérieure de Cachan, 11 2010.
- [GJ\*10] GUENNEBAUD G., JACOB B., ET AL.: Eigen v3. <http://eigen.tuxfamily.org>, 2010.
- [Gra06] GRADY L.: Random walks for image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 28, 11 (2006), 1768–1783.
- [KBH06] KAZHDAN M., BOLITHO M., HOPPE H.: Poisson surface reconstruction. In *Proceedings of EUROGRAPHICS Symposium on Geometry Processing* (2006), pp. 61–70.
- [KSO04] KOLLURI R., SHEWCHUK J., O’BRIEN J.: Spectral surface reconstruction from noisy point clouds. In *Proceedings of EUROGRAPHICS Symposium on Geometry Processing* (2004), pp. 11–21.
- [MBG\*12] MELLADO N., BARLA P., GUENNEBAUD G., REUTER P., SCHLICK C.: Growing Least Squares for the Continuous Analysis of Manifolds in Scale-Space. *Computer Graphics Forum* (July 2012).
- [MDGD\*10] MULLEN P., DE GOES F., DESBRUN M., COHEN-STEINER D., ALLIEZ P.: Signing the unsigned: Robust surface reconstruction from raw pointsets. *Computer Graphics Forum* 29, 5 (2010), 1733–1741. *Proceedings of EUROGRAPHICS Symposium on Geometry Processing*.
- [NIL12] NGUYEN C., IZADI S., LOVELL D.: Modeling kinect sensor noise for improved 3d reconstruction and tracking. In *3D Imaging, Modeling, Processing, Visualization and Transmission* (2012), pp. 524–530.
- [PGK02] PAULY M., GROSS M., KOBELT L.: Efficient simplification of point-sampled surfaces. In *Proceedings of IEEE Visualization* (2002), pp. 163–170.
- [RY07] RINEAU L., YVINEC M.: A generic software design for Delaunay refinement meshing. *Computational Geometry* (2007), 100–110.
- [Son10] SONG Y.: Boundary fitting for 2D curve reconstruction. *The Visual Computer* 26 (2010), 187–204.
- [Sot06] SOTOODEH S.: Outlier detection in laser scanner point clouds. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 5 (2006), 297–302.
- [SSZCO10] SHALOM S., SHAMIR A., ZHANG H., COHEN-OR D.: Cone carving for surface reconstruction. *ACM Transactions on Graphics* 29, 6 (Dec. 2010), 150:1–150:10.
- [TOZ\*11] TAGLIASACCHI A., OLSON M., ZHANG H., HAMARNEH G., COHEN-OR D.: Vase: Volume-aware surface evolution for surface reconstruction from incomplete point clouds. *Computer Graphics Forum* 30, 5 (2011), 1563–1571.
- [ULVH10] UNNIKRISHNAN R., LALONDE J.-F., VANDAPPEL N., HEBERT M.: Scale selection for geometric fitting in noisy point clouds. *International Journal on Computational Geometry and Applications* 20, 5 (October 2010).
- [WSS09] WANG H., SCHEIDEGGER C. E., SILVA C.: Bandwidth selection and reconstruction quality in point-based surfaces. *IEEE Transactions on Visualization and Computer Graphics* 15, 4 (2009), 572–582.